

Docket No. AT9-98-908

CLAIMS:

What is claimed is:

- 1 1. A process for invoking a method of a server object
2 in a distributed application in a distributed data
3 processing system, the process comprising the computer-
4 implemented steps of:
5 executing a client object, wherein the client object
6 is implemented in a different programming paradigm from
7 the server object;
8 obtaining an object reference for a remote server
9 object;
10 wrapping the object reference in an adapter; and
11 invoking a method of the adapter.
- 1 2. The process of claim 1 wherein the adapter uses the
2 object reference to invoke a method of a skeleton on the
3 server.
- 1 3. The process of claim 1 wherein the skeleton invokes
2 a method of a server object.
- 1 4. A method for implementing a distributed application
2 in a distributed data processing system, the method
3 comprising the computer-implemented steps of:

Docket No. AT9-98-908

4 obtaining an object reference for a proxy of a
5 server object;
6 wrapping the proxy in an adapter; and
7 invoking a method of the adapter.

1 5. The method of claim 1 wherein the adapter is a Java
2 class that implements an interface supported by the
3 server object.

1 6. The method of claim 1 wherein the server object is
2 an Enterprise JavaBean.

1 7. The method of claim 1 wherein the object reference
2 is obtained from a naming service.

1 8. The method of claim 1 wherein the proxy is a Common
2 Object Request Broker Architecture (CORBA) proxy.

1 9. The method of claim 8 wherein the adapter calls a
2 method of the CORBA proxy.

1 10. The method of claim 8 wherein the CORBA proxy is a
2 Java class residing on a client computer.

1 11. The method of claim 8 wherein the CORBA proxy passes
2 the method request to an object request broker.

Docket No. AT9-98-908

1 12. A data processing system for invoking a method of a
2 server object in a distributed application in a
3 distributed data processing system, the data processing
4 system comprising:

5 execution means for executing a client object,
6 wherein the client object is implemented in a different
7 programming paradigm from the server object;

8 obtaining means for obtaining an object reference
9 for a remote server object;

10 wrapping means for wrapping the object reference in
11 an adapter; and

12 invoking means for invoking a method of the adapter.

1 13. The data processing system of claim 12 wherein the
2 adapter uses the object reference to invoke a method of a
3 skeleton on the server.

1 14. The data processing system of claim 12 wherein the
2 skeleton invokes a method of a server object.

1 15. A data processing system for implementing a
2 distributed application in a distributed data processing
3 system, the data processing system comprising:

4 obtaining means for obtaining an object reference
5 for a proxy of a server object;

6 wrapping means for wrapping the proxy in an adapter;
7 and

8 invoking means for invoking a method of the adapter.

Docket No. AT9-98-908

1 16. The data processing system of claim 15 wherein the
2 adapter is a Java class that implements an interface
3 supported by the server object.

1 17. The data processing system of claim 15 wherein the
2 server object is an Enterprise JavaBean.

1 18. The data processing system of claim 15 wherein the
2 object reference is obtained from a naming service.

1 19. The data processing system of claim 15 wherein the
2 proxy is a Common Object Request Broker Architecture
3 (CORBA) proxy.

1 20. The data processing system of claim 19 wherein the
2 adapter calls a method of the CORBA proxy.

1 21. The data processing system of claim 19 wherein the
2 CORBA proxy is a Java class residing on a client
3 computer.

1 22. The data processing system of claim 19 wherein the
2 CORBA proxy passes the method request to an object
3 request broker.

1 23. A computer program product in a computer readable
2 medium for use in a data processing system for invoking a
3 method of a server object in a distributed application in

Docket No. AT9-98-908

4 the distributed data processing system, the computer
5 program product comprising:
6 first instructions for executing a client object,
7 wherein the client object is implemented in a different
8 programming paradigm from the server object;
9 second instructions for obtaining an object
10 reference for a remote server object;
11 third instructions for wrapping the object reference
12 in an adapter; and
13 fourth instructions for invoking a method of the
14 adapter.

1 24. A computer program product in a computer readable
2 medium for use in a data processing system for
3 implementing a distributed application in a distributed
4 data processing system, the computer program product
5 comprising:
6 first instructions for obtaining an object reference
7 for a proxy of a server object;
8 second instructions for wrapping the proxy in an
9 adapter; and
10 third instructions for invoking a method of the
11 adapter.